QLAVE

VOLUMEN I NO.4 ABRIL 1986

INFORMACION SOBRE EL CLUB

La integración en la asociación C.E.I.U.Q.L. se hace por suscripción anual o semestral. El C.E.I.U.Q.L. publica mensualmente el boletín de los socios QLave, nombre por el cual también se conoce al club.

Más información sobre la asociación se puede obtener desde la secretaría del club.

Para ser miembro de QLave se requiere estar interesado en el micro-ordenador Sinclair QL.

El club mantiene una librería de software. Una lista de los programas existentes en la librería se publicará de vez en cuando para así ir actualizándola. Los programas que se quieran aportar o sacar de la librería se deben notificar directamente al encargado de la misma.

Por favor enviad todas las contribuciones a QLave al presidente.

Presidente	Secretario	Librero
Serafín Olcoz	Juan Palacio	Angel Asín
Baltasar Gracián 21,1-cto.	Pedro M. Ric 19, 4	San Jorge 22,3-B
50005 ZARAGOZA	50008 ZARAGOZA	50001 ZARAGOZA

Contribuciones a QLave:

Las contribuciones a QLave deben ser Archivos de QUILL en cartuchos de microdrive, preferiblemente. Los cartuchos se devuelven a vuelta de correo.

Los programas cuya extensión no sea muy grande se incluirán en la revista, pero aquellos de gran extensión pasarán a formar parte de la librería.

Los programas que enviéis deben adjuntar una descripción de los mismos y de su funcionamiento.

==========

Se prohibe la reproducción total o parcial del contenido de esta revista.

QLAVE no se hace responsable del contenido de los artículos o comentarios que aparezcan firmados por sus correspondientes autores.

Sinclair, QL, QDOS, ZX microdrive son marca registrada de Sinclair Research Ltd.
QUILL, ABACUS, ARCHIVE, EASEL son marca registrada de PSION Ltd.

EDITORIAL

Nuevamente tengo que dar la mala noticia de que todavía no he recibido noticias del Gobierno Civil referentes a los estatutos de Glave. Ante la falta de novedades y la tardanza de las mismas me pasé en persona por las dependencias del registro de asociaciones del Gobierno Civil de Zaragoza para preguntar por la "salud" de nuestros estatutos y allí me dijeron que junto con otros estaban retenidos en Madrid y que los habían reclamado por lo que esperaban recibirlos pronto. Cada vez comprendo mejor la subjetividad de la palabra pronto.

Como decía en mi anterior editorial tengo que dar la enhorabuena a los socios que empiezan a sacarle el jugo y a darle vida a Glave, esto es a aquellos que colaboran en la sección cartas de los socios y en la de preguntas así como también en la librería de programas que parece que por fin va a arrancar.

Aprovecho estas líneas para agradecer las críticas que me habéis hecho llegar por carta con respecto al boletín espero que lleguen más y así poder mejorar, ánimo y muchas gracias de nuevo.

De todas formas y aunque parece que ya se va rompiendo la inercia, tengo que hacer un nuevo llamamiento a todos los socios para que dejéis (me refiero a la mayoría de vosotros) la situación pasiva, receptora frente a Glave y que colaboréis en las secciones que más os apetezcan y si os hay alguna de vuestro agrado escribidme con vuestras sugerencias ya que para eso estamos.

Serafín Olcoz

COMENTARIO DE PROGRAMAS

PROGRAMA: ALSICONT
EDITORES: ALSISA
DIRECCION: ANTONIO LOPEZ, 117 2-B 28026 MADRID

El programa se entrega en dos cartuchos y una copia de seguridad asi como con las instrucciones pertinentes. En un cartucho está la parte básica del programa en el otro tenemos unos ficheros con el plan de cuentas ya creado pero que no es necesario si lo que deseamos es crearnos un plan a nuestra medida.

El formato en que aparece el programa es el habitual en todos los "alsis", una ventana superior donde se muestran una serie de comandos básicos y la pantalla principal donde se ejecuta el programa. Al comenzar se nos pide colocar en hora y fecha el reloj interno que es sacado por pantalla de forma similar al que conocemos del manual de Sinclair y de Toolkit.

El menú principal consta de las siguientes opciones: Plan de cuentas- Diario de movimientos- Menú de totales- Regularización, cierre, reapertura- Reposición de saldos- Preparar la contabilidad y Copias.

Al comenzar la ejecución del programa por primera vez comenzaremos por preparar la contabilidad donde se nos pregunta por el nombre de la empresa, los baudios a los que funciona nuestra impresora y el número de lineas por página. No se indica nada del número de columnas por linea ya que el programa toma invariablemente el comprimido que es el único formato que permite imprimir correctamente todo en una impresora de 80 columnas.

Nuestro siguiente paso será incluir en nuestro plan de cuentas aquellas que vayamos a utilizar sin olvidar abrir las correspondientes subcuentas ya que es con ellas con las que trabaja el programa de forma directa y las cuentas sólo son usadas para realizar la acumulación de su grupo. Asi en la opción del menú principal podremos agregar cuentas y subcuentas al plan, borrarlas y obtener impresión del plan tanto por pantalla como por impresora.

Ahora ya estamos en condiciones de comenzar la introducción de los asientos a la que se accede por la opción número dos del menú (Diario de movimientos). La introducción se realiza por el método partida - contrapartida, es decir, en primer lugar se teclea la anotación a realizar en el debe y en el mismo momento se nos pregunta por la contrapartida pudiendo ser ésta lógicamente múltiple, es decir varias contrapartidas que deberán sumar lo mismo que el primer apunte realizado. En cualquier momento dentro de este submenú dos podemos solicitar la impresión del diario acumulado que aparece en un formato agradable y bien estructurado. Las otras opciones son las de borrar asientos y retorno a menú principal.

PASIVO AL __111111

Cuenta	Titulo	Saldo Cuenta	Saldo Grupo
100	capital social	-500000	
113	Reserva Legal	-138545	
130	Remanente	0	
.....CAPITAL Y RESERVAS			-638545
.....SUBVENCIONES EN CAPITAL			0
.....PREVISIONES			0
490	Prov.por Insol.	-126824	
.....PROVISIONES			-126824
171	Prestam. Red.plazo	-4899617	
.....DEUDAS A MEDIO Y LARGO PLAZO			-4899617
400	Proveedores	51274	
420	Efec,Comer./pagar	-529367	
438	Partid. Pend. apl.	0	
475	H.P.Acreedores I.G.T	-46302	
477	Org. S.S.acreedor	0	
476	H.P.Acreedoresl.R.P.	22902	
.....DEUDAS A CORTO PLAZO			-503493
451	Comisiones Pendiente	-744439	
485	Pagos Diferidos	0	
.....AJUSTES POR PERIODIFICACION			-744439
969	Perdidas y Ganancias	0	
.....RESULTADOS			0
TOTAL PASIVO.....		-6003918	-6003918

ACTIVO AL __111111

Cuenta	Titulo	Saldo Cuenta	Saldo Grupo
203	Maq. Ins. Util.	471534	
204	Element. Trans.	0	
205	Mobil y Enseres	1166350	
208	Otros Inmov. Mat.	0	
260	Fianzas a largo	0	
270	Gastos de Const.	0	
280	Amortiz.Acum.	-91640	
.....INMOVILIZADO			1546244
300	Exis. Mercad.	2420261	
390	Prov. depre. exist.	-151240	
.....EXISTENCIAS			2269021
430	Clientes	0	
435	Clientes dud. cob.	0	
450	Efectos a Cobrar	1791401	
455	Efectos Impag	0	
490	Prov.por Insol.	-126824	
.....DEUDORES			1664577
570	Caja Pesetas	-97240	
571	B Popular 50215-22	1000	
572	Banco Central	498480	
573	Bankinter	46	
574	C.A.R.P.	-6181	
575	Banesto	141999	
576	Cuentas de Ahorro	21388	
.....CUENTAS FINANCIERAS			559494
.....SITUAC. TRANS. DE FINANC.			0
480	Pagos Antic.	0	
486	Cobros Diferidos	0	
.....AJUSTES POR PERIODIF.			0
800	Perdidas y Ganancias	0	
.....RESULTADOS			0
TOTAL ACTIVO.....		6033336	6033336

La tercera opción principal nos lleva a un extenso submenú de totales que nos permite obtener la siguiente información:

- Activo y Pasivo perfectamente ordenado por grupos según el Plan general de Contabilidad y acumulado por cuentas
- Estado de cuentas y subcuentas
- Balance de sumas y saldos, opción necesaria y que nos indicará de cualquier error cometido en la introducción de los apuntes y que llevase a descuadre de éste, esto es de agradecer pero quizá el programa debiera tener más recursos a lo largo de su ejecución para no tener que llegar hasta aquí para detectar el fallo.
- Obtención de extractos de cuentas y subcuentas, que se diferencia de la obtención del estado en que éste se refiere a totales y los extractos dan detalle de los apuntes introducidos.
- Impresión del Mayor de cuentas
- Actualización de los asientos, que se realizará si deseamos consolidar los resultados de las cuentas y liberar de espacio al diario.

La opción quinta del menú principal se refiere a la reposición de saldos, en la que se pueden indicar nuevos saldos iniciales para todas las cuentas. Esta es una opción normal y obligatoria la primera vez que pongamos en marcha el programa para situar al día nuestra contabilidad. En ejercicios sucesivos todo esto ya será realizado automáticamente por el programa en su opción sexta, que se ocupa de realizar la regularización del ejercicio obteniendo la cuenta de explotación y la de pérdidas y ganancias, su cierre y la reapertura de un nuevo ejercicio.

El programa está preparado para formar parte del paquete de programas Alsipack que recoge el grueso de las aplicaciones necesarias en la gestión de la empresa.

Destacar del programa la gran cantidad de apuntes realizables, así un microdrive permite 3000 y un disco de 720 k. unos 40000. La presentación es bastante agradable y suministra toda la información que se le puede exigir a este tipo de programas.

En los defectos achacables al programa podemos referirnos a que el programa no toma fecha por defecto, así siempre es preciso teclearla cuando se nos solicita si bien el programa ya debería saberla al haber colocado en hora el reloj interno al comienzo del programa. Y respecto a las fechas decir también que acepta cualquier tipo de fecha, así no comprobará si este año es posible la existencia de un 29 de Febrero e incluso tragará un 35 de Enero.

Existen pocos mensajes de error a lo largo del programa y los que aparecen son demasiado fugaces en su aparición por la pantalla. Esta deficiencia de mensajes de error lleva al programa a que se pueda colgar con frecuencia si no se tiene cuidado al teclear los datos ya que una respuesta de cadena vacía (enter) a un requerimiento de tipo input del programa puede provocar el cuelgue de éste y lógicamente si no se habían introducido por ejemplo los asientos en el microdrive éstos se perderán y será necesario recomenzar su introducción aunque en este programa no suceda como con los programas de Psion que era

necesario cerrar todos los ficheros de forma expresa, de manera que aunque el programa se nos corte todos los datos que hayamos guardado en el microdrive seguirán siendo accesibles. Siguiendo en el problema de la falta de mensajes de error decir que por ejemplo el programa se cortará si introducimos una cantidad muy elevada .

Ahora bien todos estos errores son fácilmente subsanables y suponemos que en futuras versiones del programa podrán ser corregidos.

A la hora de exigirle más prestaciones al programa se podía haber incluido la posibilidad de variar los formatos de salida de la información o que los extractos de las cuentas se realizasen entre fechas determinadas y no un extracto indiscriminado como se hace en el programa.

Decir por último que tal vez el programa esté demasiadas veces accediendo a los microdrives, ya que lo hace prácticamente en todo momento y esto si se usa disco no es excesivamente grave pero sí que retrasa mucho el trabajo con microdrives.

Así pues podemos decir que el programa está bastante bien en las facilidades ofrecidas para la contabilización pero posee algún fallo como los comentados que exigen una atención especial por el usuario del programa.

@Lave.

EXTRACTO DE LA SUBCUENTA 700001_Ventas eje. transp.

NU	FECHA	JUSTIF	CONCEPTO	IMPORTE	SALDO
1	3106	450007	Nasa	2739	h
1	3106	450009	Urola	268836	h
1	3106	450005	Urola	115216	h
1	3106	450009	erorlasa	132924	h
163	0907	450010	Aceites Toledo	13010	h
225	2307	450010	Vanguardia	37487	h
231	2407	450010	Nasa	57157	h
353	2407	572001	Doja(30%)	41003	h
517	1209	450012	Doja	96631	h
1	1209	450012	El Qexigal	186671	h
521	1209	450011	IQAP	83267	h
523	1209	450010	Iberfrisa	2430	h
1	1809	450012	Pasa	148237	h
1	1809	572001	30% Orbelan	330580	h
1	2509	450012	Kanfort	141277	h
1	2509	450010	Direccion Trafico	836755	h
673	3009	450010	Pycasa	9035	h

SALDO INICIAL	-7355413
SALDO ACUMULADO	519715
TOTAL DE DIARIO	-2499281
TOTAL DE LA SUBCUENTA	-9334979

DIARIO DE MOVIMIENTOS AL __111111

NU	Fecha	Cuenta	Justif	Concepto	Importe	Saldo
1	3106	450007	700001	Nasa	2739	d
2	3106	700001	450007	Nasa	2739	h
3	3106	450009	700001	Urola	268836	d
4	3106	700001	450009	Urola	268836	h
5	3106	450005	700001	Urola	115216	d
6	3106	700001	450005	Urola	115216	h
7	3106	600002	420009	Zubiri	6697	d
8	3106	420009	600002	Zubiri	6697	h
9	3106	600001	420009	Rutllo	53942	d
10	3106	420009	600001	Rutllo	53942	h
11	3106	600001	420009	La Platina	3161	d
12	3106	420009	600001	La Platina	3161	h
13	3106	300001	575001	Urena	9275	d
14	3106	575001	300001	Urena	9275	h
15	3106	575001	450004	Inma	6372	d
16	3106	450004	575001	Inma	6372	h
17	3106	645001	575001	J.Ortega	15150	d
18	3106	575001	645001	J.Ortega	15150	h
19	3106	575001	700002	programa	18420	d
20	3106	700002	575001	programa	18420	h
21	3106	620001	572001	remesa	210	d
22	3106	572001	620001	remesa	210	h
23	3106	620001	572001	devolucion	1571	d
24	3106	572001	620001	devolucion	1571	h
25	3106	620001	572001	remesa	740	d
26	3106	572001	620001	remesa	740	h
27	3106	640001	570001	Zurita	2354	d

NOVEDADES

QL Bright Star Modem (con interfaz para impresora)

Auto Answer Modem 179 £ Auto Answer / Auto Dial Modem 209 £

Especificacioes :

CCITT V21/23, BELL 103/113 300/300, 1200/75, 75/1200, 1200/1200
Todo esto más : Interfaz Centronics para impresora con buffer de 2K, y cable para QL.
Incluido software "STAR TERM" - emulación PRESTEL, emulación de terminal de 40 ó 80 columnas, facilidad de transferencia de archivos y muchas otras caracteristicas extras.

Inteligente :

Caracteristicas de Auto scan sitúan al modem en la forma correcta de respuesta al portador de entradas. Control Total sobre todos los aspectos de las operciones por medio de treinta comandos de software que no es necesario aprender (el software se ocupa de ello). Incorpora menú de ayuda.

QL Programadores de EPROM y accesorios

Q-PROM Programas 2764 (8k byte), 27128 (16K byte) y versiones "A"	99	£
BLOPROM-RS Programador profesional de EPROM (conectado via RS232)	172.44	£
MULTEPROM Programador en grupo - Realiza 8 copias simultáneamente	229.95	£
Q-CART Tarjeta de expansión de ROM para QL. Para 8k o 16k EPROM	7.95	£
27128 250ns 16k byte EPROM (en blanco) Para usar con Q-CART	4.99	£
DHOBI 2 Borra hasta 3 EPROMs a la vez	29.95	£

SISTEMAS MIRACLE EXPANDARAM

256 K por 95 £ 512 K por 125 £

El Miracle expandaram es una tarjeta interfaz separada que a diferencia de otras expansiones de RAM, tiene un segundo conector, y permite conectar otro interfaz como por ejemplo el de Cumana Floppy disc.

BOOT 128K

5.99 £ en Microdrive, esta utilidad es para usarla en QLs con más memoria que los 128 K RAM standard. Permite utilizar software que no funcionaría normalmente en un QL expandido. También incluye : BACKUP y PURGE (para microdrives y floppy disc drives).

COMPWARE OFERTAS ESPECIALES	PRECIO	PRECIO DE OFERTA Gastos de envio incluidos
Interfaz de Disco Micro Peripherals / Sinclair	114 £	75 £
Expansión Simplex 256 K RAM	99 £	75 £
Project Planner & Decision Maker (Juntos)	79 £	69 £
Backgammon	15 £	12 £

Fundas vacias de Microdrive : 1 £ cada una / 2 £ si son 4 / 4,5 £ si son 10 fundas

QL Floppy Disc Drives con Expansión de Memoria	Con UNA unidad de Discos CUMANA	Con DOS unidades de Discos CUMANA
Interfaz Cumana (Sin RAM)	230 £	310 £
Interfaz Delta (128K RAM e interfaz de impresora)	340 £	420 £
Interfaz PCML con 256K RAM	380 £	460 £
Interfaz Cumana + 256K Miracle Expandaram	325 £	405 £
Interfaz Cumana + 512K Miracle Expandaram	355 £	435 £

Caja de 10 DSDD floppy discs 3.5" (SONY, CUMANA, o MAXELL) 49 £

Caja de 10 DSDD 3.5" sin marca 3M 39 £

La forma de realizar la compra de los productos que ofrece COMPWARE está descrita en el número 3 de QLave.

A lo ya dicho en el pasado número hay que añadir que si se utiliza una transferencia bankaria hay que añadir 10 £, que es lo que cargan los bancos de U. K. por sus servicios.

Respecto al coste que hay que añadir por gastos de envio, la nueva lista es:

Software de Juegos, Microdrives, Cables	2 £
Software de Computer One / Adder	3 £
Disc interfaces, RAM expansiones, Centronics i/f	4 £
Software de Melacomco y Sinclair, Busnss. y Técnico	5 £
CST Q+4, MODEMS	12 £
Sinclair QL, Disc Drives	10 £
Impresoras, Monitores	25 £

COMPWARE

Hardware

Sinclair QL (Psion 2 packages/ 5 free games)..... £199-99

Q-MOD £79 Q-CON £89 Q-CALL £50

Microvitec CUB 14" high res. colour monitor with cable & plynth. £289-00
Phillips Monochrome 12" monitor with cable: Green screen... £109-99
(Tilt plynth for Phillips £9.99) Amber screen... £114-99

Q+4, 4 slot expansion module (CST)..... £199-99
Expansion module (Sinclair QL) power supply..... £ 25-00

Miracle Expandaram with "through connector"
for Cumana floppy disc interface: 256K..... £ 95-00
512K..... £125-00

Cumana disc interface (The best yet!)..... £ 89-00
Delta disc interface (parallel printer i/f)..... £129-00
Delta disc interface (128K RAM, printer i/f)..... £199-00
PCML disc interface with 256K RAM..... £239-00

3.5" 720K single drive (built in psu)..... £141-00
3.5" 720K dual drives (built in psu)..... £221-00

CP80 80 column dot matrix printer..... £225-00
Kaga Taxan Letter Qality Matrix printer..... £309-00
Daisy Step 2000 Daisywheel printer (used to print $a^2 - b^3$ this!) £279-00

3.5" floppy (DS/DD) (Sony, Cumana or Maxell) 1=£5-49 (box of 10) £ 49-00
3.5" floppy (DS/DD) unbranded 3M.....(box of 10) £ 39-00

4 microdrives in plastic wallet (as with QL)..... £ 7-99
QL serial cable or Medic Printer cable..... £ 15-00
QL Centronics interface..... £ 38-00

Business Software

Cash Trader..... £69 Entrepreneur..... £39
Project Planner..... £39 Decision Maker..... £39

Software Languages & Utilities

QL MDV Toolkit (Extensions) £ 9-99 QL Assembly Lang MDV Toolkit. £29-99
Sinclair/Tony Tebby Toolkit £24 Sinclair Technical Guide..... £14
Supercharge (Basic compiler)£59 Computer One Assembler..... £29
QDRAW (Psion)..... £15 Metacomco Assembler..... £39
Sprite Designer (Snowsoft). £ 9 Adder QL Assembler..... £34
Metacomco C..... £99 Metacomco BCPL..... £59

Software Games

QL Backgammon..... £12 Hopper (Microdeal)..... £14
Hungry Harry..... £10 Psion Chess..... £18
Psion Match Point..... £18 QL Hyperdrive (RRRacing)..... £14
QL Caverns (Sinclair)... £14

PRICES INCLUDE VAT AND DELIVERY
WRITE OR TELEPHONE FOR FURTHER DETAILS OR GENERAL HELP

UK residents: Add £2.50 per order if you require Cash On Delivery but note that your total order value must not exceed £347.50 to use the COD service.

PREGUNTAS

GRABACION DE DATOS DESDE UN PROGRAMA

Anteriormente yo era usuario de ZX-Spectrum y para grabar una matriz llena de datos del tipo DIM A$ (100,30) yo hacia SAVE#"M";1;"matriz" DATA A$(), y después podia utilizarla para cualquier programa. ¿Como puedo hacerlo con el QL!!!?

José M. Tejero
CEUTA

El sistema de grabación de datos del Spectrum es completamente distinto al usado por el SuperBasic. El QL dispone de comandos como: OPEN, OPEN_IN, OPEN_NEW, PRINT #n, INPUT #n... que permiten la gestion de ficheros desde un programa de forma más estructurada.

A continuación mostramos dos programitas que explican por si solos la manera de operar. El primer listado, crea un fichero de 5 datos alfanumericos, y los graba en el microdrive 2 con el nombre prueba (linea 50).

El segundo listado vuelve a formar la matriz de los 5 nombres, tomándolos del microdrive 2.

```
10 DIM A$(5,10)
20 FOR N=0 TO 5
30 INPUT A$(N)
40 NEXT N
50 OPEN_NEW #4, MDV2_PRUEBA
60 PRINT #4, A$
70 CLOSE #4
```

```
10 OPEN #4, MDV2_PRUEBA
20 DIM A$(5,10)
30 FOR N=0 TO 5
40 INPUT #4, A$(N)
50 NEXT N
```

En la Guia del Usuario del QL, se hace referencia a este tema al comentar los comandos PRINT, INPUT, OPEN; y en el capitulo dedicado a "canales", de la guia de referencia de conceptos.

QLave.

Desearía saber si en el programa "ARCHIVE", cuando trabajas con varios ficheros pertenecientes al mismo asunto, y que tuvieran algún campo común, por ejemplo el nombre y apellidos, existe algún procedimiento por el que se pueda introducir el contenido de una sola vez y que aparezca en todos los ficheros y si haces una modificación en el contenido de un campo, se cambien los campos homónimos de los demás ficheros.

Por último si alguien sabe las diferencias que hay en hacer una ventana con scr_ o con_, o con window o con block.

José Aramendi Lizcano
MADRID

ARCHIVE es una base de datos que posee un lenguaje de programación propio. Dicho lenguaje tiene sus funciones (pags. 52 a 55 del manual) y comandos (pags. 44 a 52 del manual). Estos pueden emplearse aisladamente (ej: indicar), o bien formando procedimientos. Para llevar a cabo tareas simples como formatear un cartucho, abrir un fichero para insertar un nuevo registro, resultaría tedioso el hacer un procedimiento, y es más cómodo hacerlo en "modo comando". Por el contrario tareas como la descrita en la pregunta pueden llevarse a cabo en "modo comando" (alterando primero los campos de un fichero, buscando luego los campos homónimos en otro fichero, para llevar a cabo la misma operación), pero no resulta aconsejable si es algo que se debe realizar con cierta frecuencia; en este caso resulta más practico elaborar un procedimiento, en el que estén descritos todos los pasos que se deben llevar a cabo con los ficheros que se desee. Dicho procedimiento puede grabarse en un cartucho, y cada vez que deseemos hacer la tarea en cuestión bastará con cargarlo y llamarlo. El se preocupará de preguntarnos los datos y registros con los que va a trabajar y de hacer las actualizaciones pertinentes (según el trabajo para el que lo hallamos diseñado).

La manera de elaborar un procedimiento que se ajuste a nuestras necesidades se describe en el capitulo 8 de la sección de la Guia de Usuario dedicada a ARCHIVE, junto con ejemplos de procedimientos desarrollados para aplicaciones concretas

---

BLOCK Y WINDOW son dos comandos de SuperBasic, mientras que scr_ y con_ son dos nombres de dispositivos, igual que lo son mdv1_ o ser2_.

scr_ y con_ (screen y console), ofrecen posibilidades muy similares para trabajar, asignando una ventana de la pantalla (dispositivo externo), a un canal. La diferencia entre ambos estriba que scr se usa como dispositivo exclusivo de salida de datos, por lo que resultara imposible hacer un INPUT en un canal asignado a scr_, cosa que no ocurre con con_.

Así pues puede parecer que las lineas siguientes llegan al mismo resultado:

```
10 OPEN #4, con_100x100a0x0
20 OPEN #5, scr_100x100a0x0
```

Sin embargo, y siguiendo el ejemplo, no sería válido "INPUT #5, A$".

Los parámetros para la definición de la ventana, en ambos casos se dan en pixels según el formato:

scr_(anchura)x(altura)a(coordenada x)x(coordenada y)

con_(anchura)x(altura)a(corrdenada x)x(coordenada y)_(longitud del buffer del teclado)

WINDOW es un comando Basic que permite cambiar la posición y tamaño de una determinada ventana ya existente; mientras que con scr_ y con_ creamos una nueva ventana asignàndola a un canal.

Por ejemplo, una vez creada una ventana : OPEN #4, con_200x200a10x10 , al teclear: WINDOW #4, 100,100,0,0 cambiamos las dimensiones y la ubicación.

BLOCK se limita a dibujar un bloque en una ventana, de las dimensiones que especifiquemos.

@Lave.

Respecto a la pregunta sobre sistemas operativos existentes para el Q.L. además del ya conocido QDOS existen:

- Un CPM/80 creado por PCW L.
- El 68k/OS de GST.
- El CP/M 68K (versión para el micro MC68000) de QUEST.

Y los desarrollados para acoplar ratón al Q.L.

¿Qué posibilidades hay de lograr la impresión de los gràficos de EASEL en una impresora Seikosa 500?

¿Qué es lo que hay que hacer para imprimir un gràfico u otra cosa de la pantalla?

José Andrés Sagarra Miró
SALAMANCA

De tu carta deducimos que no has logrado hacer copys de pantalla y ello puede ser debido a que si estas conectado en serie a tu impresora no hayas colocado los baudios a 4800.

Junto con el Easel se incluye una rutina en código màquina para imprimir los gràficos que en la versión española se llama GPRINT_PRT y que se diferencia de los demàs ficheros con extensión _PRT en que cada una de las impresoras a las que se refiere tiene distinto código de control de Bit Image el estàndar incorporado en GPRINT_PRT es (ESC), L, long. en bytes por línea. Prueba seleccionando cada uno de los distintos tipos de impresora desde EASEL y si no te funciona ninguno entonces salva el fichero mediante la opción imprimir en microdrive y usa la rutina que incluimos en este mismo número debidamente modificada con los códigos de control para Bit Image que son el 8 (Backspace en las demàs impresoras) y luego el número de bits por línea que en tu caso son 640 (en las demàs se suele usar 1280).

TRUCOS Y RUTINAS

NOMBRES DE FICHEROS

En la guia del usuario se indica la siguiente sintaxis para especificar un nombre de fichero:

MDVn_(nombre de fichero) --siendo n el número de microdrive--

Pero no se aclara la posibilidad de sustituir dicho nombre por una variable alfanumérica.

Supongamos una rutina que tenga que grabar un programa Basic o una zona de memoria, un fichero... y que pueda preguntar al operador el nombre con el que lo va a grabar. Presentaria una configuración similar a esta:

```
1000 INPUT "Nombre para grabar datos", A$
1010 PRINT "Introduzca un cartucho formateado en MDV1_ y pulse una tecla"
1020 IF INKEY$="" THEN GOTO 1020
1030 SBYTES MDV1_A$, 131072,32768
```

En principio, la linea 1030 parece correcta, y la guia del usuario no explica otra forma de hacerlo, pero el interprete Basic no la ejecutará y devolverá el error "expresión errónea".

Al asignar un nombre de fichero contenido en una variable, debe entrecomillarse el nombre del microdrive y unirle (&) la variable que deseemos usar:

```
SBYTES "MDV1_"&A$, 131072,32768
        Similarmente
SAVE "MDV1_"&A$  /  LOAD "MDV1_"&A$
```

El usar una variable en vez del nombre, permite además usar nombres no permitidos por el interprete SuperBasic. Nombres como "01/03/86/" o "mi fichero", o " " no son licitos y rechazaria las instrucciones:

```
SAVE MDV1_mi fichero
SAVE MDV1_          (uno o varios espacios)
```

Sin embargo aceptaria:

```
A$="mi fichero" : SAVE "MDV1_"&A$
A$=" " : SAVE "MDV1_"&A$
```

E Igualmente :

```
A$=" " : LOAD "MDV1"&A$
```

Incluso se puede conseguir grabar y cargar programas sin nombre, asignando A$="" y luego SAVE "MDV1_"&A$, o simplemente SAVE MDV1_. Luego podrian cargarse de forma similar. Este puede ser un metodo para evitar que aparezca el nombre en el directorio del microdrive.

P R O G R A M A S A U T O E J E C U T A B L E S

Si creamos un fichero que constituya un programa Basic, sin números de linea, luego podremos cargarlo igual que se de un programa normal se tratara, con la salvedad de que se ejecutará automáticamente, y no quedará el listado.

Prueba con el siguiente Ejemplo:

```
10 OPEN_NEW #4, MDV1_prueba
20 PRINT #4, "WINDOW 512,256,0,0"
30 PRINT #4, "CLS"
40 CLOSE #4
```

Introduce el cartucho (que no tenga ningún fichero llamado "prueba"), y ejecutalo.

Ahora preba a cargar el programa "prueba" que te ha grabado en el microdrive ¡ sin que se ejecute.

@Lave.

El siguiente programa en basic crea una rutina en lenguaje maquina que se grabara en el segundo microdrive en formato 'sbytes', para cargarlo por medio de la instruccion 'lbytes'. El proposito de la rutina es el de crear un procedimiento residente, que se llamara con el nombre de 'dump' (sin parametros) que cree a su vez un job que haga (en multitarea) un volcado de la pantalla a impresora.

Tras ejecutar este programa que viene a tardar algo mas de medio minuto, para cargar el programa en codigo maquina solo se debera poner algo asi:

```
NN=RESPR(460)
LBYTES MDV2_DUMP_BYT,NN
CALL NN
```

A partir de este momento, cuando se escriba 'dump' se ejecutara la rutina.

La rutina se reserva para ella 32k de memoria, y en caso de que no este disponible dicha cantidad, suspende al basic.

El tiempo aproximado de ejecucion es de quince minutos, aunque aumentara sensiblemente si se esta trabajando mientras en basic y con microdrives u otros perifericos.

Los baudios los coloca la propia rutina a 4800, en caso de que sea la comunicacion a 9600 baudios hay que cambiar el valor 12C0 de la linea 530, duodecimo octeto, por el valor 25C0 (9600 en decimal).

El programa es el siguiente:

```
100 zona=RESPR(460)
110 zon=zona
120 RESTORE
130 dig$="":check=0
140 FOR tt=1 TO 20
150  READ cadena$:IF cadena$=="fin" then bytes:SBYTES mdv2_dump_byt,zona,460:STOP
160  FOR gg=1 TO LEN(cadena$)
170   IF cadena$(gg)=" " THEN
180    numero=0
190    FOR hh=1 TO LEN(dig$)
200     ascii=CODE(dig$(hh)):check=check+ascii*gg
210     IF ascii)57 THEN ascii=ascii-7
220     ascii=ascii-48
230     numero=numero+ascii*16^(LEN(dig$)-hh)
240    END FOR hh
250    dig$=""
260    POKE_W zon,numero:zon=zon+2
270   ELSE
280    dig$=dig$&cadena$(gg)
290   END IF
300  END FOR gg
310  READ checksum:IF checksum()check THEN PRINT "ERROR en la linea ";500+(t-1)*10
320  check=0
330 END FOR tt
340 DEFINE PROCEDURE bytes
350 FOR nn=1 TO 32
360  READ dato
370  POKE zon,dato:zon=zon+1
380 END FOR nn
390 RETURN
400 END DEFINE
500 DATA '3478 110 43FA 10 4E92 72FF 7600 7005 4E41 7000 4E75 1 E 444 554D 5000 0 0 0
7200 ',138828
510 DATA '7410 93C9 7001 4E41 2640 36FC 4EF9 43FA 10 26C9 7460 7600 7008 4E41 7000
4E75 223C 0 8016 74FF ',205047
520 DATA '7018 4E41 4A40 6624 2040 01FC 0 1C 267C 2 0 2A3C 0 8000 10DB 51C8 FFFC 2A4C
D9FC 0 ',159988
530 DATA '1C 6010 287C 2 0 2A4C 7400 7200 7008 4E41 323C 12C0 7012 4E41 72FF 7602 41FA
114 7001 4E42 ',186717
540 DATA '4A40 6600 D4 2C3C 0 0 243C 0 2 363C FFFF 43FA F3 7007 4E43 243C 0 7 76FF
43FA ',138267
550 DATA 'EC 7007 4E43 3A3C 100 5345 2805 C8FC 80 2644 D7C6 D7CC 1213 76FF 7005 4E43
7005 4E43 7005 4E43 ',200317
560 DATA 'BA3C 0 660C 243C 0 D 43FA BB 7007 4E43 3A3C 100 5345 2805 C8FC 80 2644 D7C6
D7CC 5248 ',173109
```

```
570 DATA '1213 363C FFFF 7005 4E43 7005 4E43 7005 4E43 BA3C 0 66D8 7401 43FA 76 76FF
7007 4E43 DC7C 2 ',195624
580 DATA 'BC3C 80 6600 FF66 7402 76FF 43FA 7A 7007 4E43 7002 4E42 B8FC 2 0 670E 204D
7019 4E41 72FF ',187842
590 DATA '7600 7005 4E41 7420 7200 7008 4E41 60EE 7424 76FF 43FA 8 7007 4E43 60C2 4572
726F 722C 2063 616E ',215945
600 DATA '61aC 2064 6520 6C61 2069 6D70 7265 736F 7261 206F 6375 7061 646F ',98795
610 DATA 'FIN'
620 DATA 10
630 DATA 27,71
640 DATA 0,0,4,83,69,82,49
650 DATA 27,65,0,27,76,0,3
660 DATA 27,72,27,51,1,10
670 DATA 27,65,0,27,76,0,3
680 DATA 27,64
```

Hay que escribir todas las datas con maysculas, y teniendo mucho cuidado de dejar siempre un espacio al final de cada cadena, antes de cerrar las comillas.

Las ultimas datas estan en forma de bytes pues son los caracteres de control que se mandan a la impresora. Esta probado en una ADKATE DP-100 (compatible EPSON FX-80).

La linea 620 contiene el codigo del retorno de carro, la 630 los codigos de 'double strike', la 640 es la definicion de canal para codigo maquina (no se envia a la impresora), la 650 contiene primero una colocacion del salto de linea a 8 inch, y coloca el 'bit image mode' (27,76,0,3) o modo de alta resolucion. 27,76 colocan el modo; y 0,3 indica la cantidad de bytes a mandar. En la linea 660, el 27,72 desactiva el modo 'double strike' (doble pasada), los codigos 27,51 anulan el incremento en el salto de linea, y el 10 es el codigo de salto de linea. La linea 670 es similar a la 650. En la 680 se dan los codigos de reinicializacion de la impresora (ESC,@).

```
MT_FRJOB        EQU         $05
IO_OPEN         EQU         $01
IO_CLOSE        EQU         $02
IO_SSTRG        EQU         $07
IO_SBYTE        EQU         $05
MT_BAUD         EQU         $12
MT_ALCHP        EQU         $18
MT_RECHP        EQU         $19
MT_CJOB         EQU         $01
BP_INIT         EQU         $110
MT_PRIOR        EQU         $0B
```

```
*
QDOS        MACRO
            MOVEQ       #\1,D0
            TRAP        #\2
            ENDM
FIN         MACRO
            MOVEQ.W     #2,D2
            MOVEQ       #-1,D3
            LEA.L       REINICIO,A1
            QDOS        IO_SSTRG,3
            QDOS        IO_CLOSE,2
            CMPA.L      #131072,A5
            BEQ.S       BASIC
            MOVE.L      A5,A0
            QDOS        MT_RECHP,1
SM          MOVEQ       #-1,D1
            MOVEQ       #0,D3
            QDOS        MT_FRJOB,1
BASIC       MOVEQ       #32,D2      En caso de haber suspendido al
            MOVEQ       #0,D1             basic, lo reactivamos
            QDOS        MT_PRIOR,1
            BRA.S       SM
            ENDM
*
* FIN MACROS
*
            MOVE.W      BP_INIT,A2    Creamos el procedimiento
            LEA.L       PROC_DEF,A1   definido en proc_def
            JSR         (A2)
            MOVEQ       #-1,D1
            MOVEQ       #0,D3
            QDOS        MT_FRJOB,1
            MOVEQ       #0,D0
            RTS
PROC_DEF
            DC.W        1
            DC.W        INICIOPROC-*  Direccion de comienzo
            DC.B        4
            DC.B        'DUMP'
            DC.W        0,0,0
INICIOPROC
            MOVEQ       #0,D1
            MOVEQ       #16,D2
```

```
SUBA.L      A1,A1
DOS         MT_CJOB,1     Avisa del nuevo job
MOVE.L      A0,A3
MOVE.W      #$4EF9,(A3)+  Instruccion de jump
LEA.L       JOB,A1        a 'job'
MOVE.L      A1,(A3)+
MOVEQ       #96,D2        prioridad 96
MOVEQ       #0,D3
QDOS        MT_PRIOR,1    Activa el job
MOVEQ       #0,D0
RTS
JOB
            MOVE.L        #$30016,d1
            MOVEQ.L       #-1,D2
            MDOS          MT_ALCHP,1    reserva 32k de memoria
            TST           D0
            BNE.S         SINMEMORIA
            MOVE.L        A0,A4
            ADDA.L        #$1C,A0
            MOVE.L        #131072,A3
            MOVE.L        #$8000,D5
SALVAR
            MOVE.B        (A3)+,(A0)+   Mueve la pantalla a la
            DBRA          D5,SALVAR     memoria reservada
            MOVE.L        A4,A5
            ADDA.L        #$1C,A4
            BRA.S         MAUDIOS
SINMEMORIA
            MOVE.L        #131072,A4    Si no hay memoria se utiliza
            MOVE.L        A4,A5         directamente la memoria de
            MOVEQ         #0,D2         pantalla,
            MOVEQ         #0,D1
            QDOS          MT_PRIOR,1    suspendiendo al basic.
```

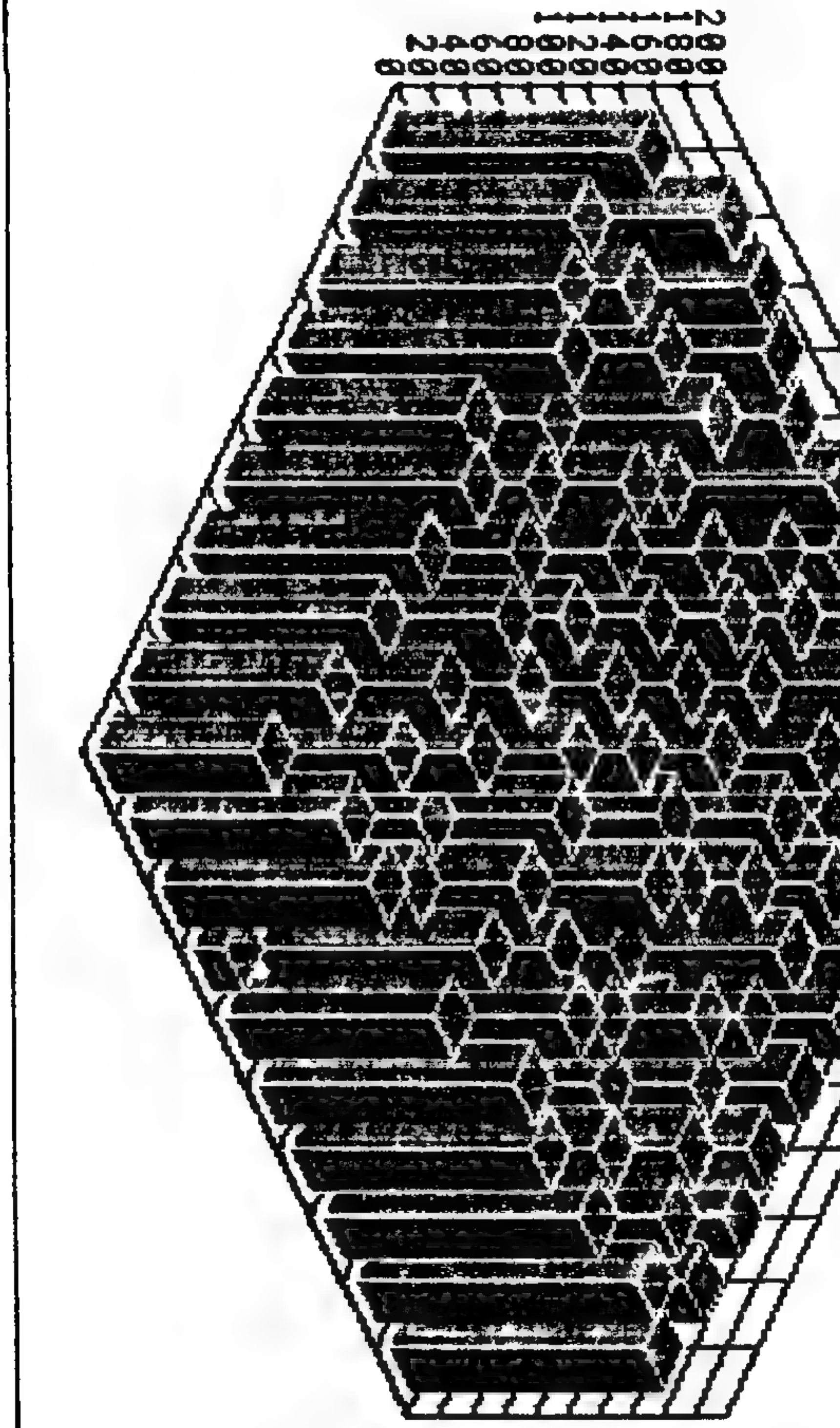
```
BAUDIOS
                MOVE.W      #4800,D1        colora los baudios a 4800
                @DOS        MT_BAUD,1
                MOVEQ       #-1,D1
                MOVEQ       #2,D3
                LEA.L       PERIFERICO,A0
                @DOS        IO_OPEN,2       y abre un canal de impresora
                TST         D0
                BNE         BUSY
                MOVE.L      #0,D6

MANDAR
                MOVE.L      #2,D2
                MOVE.W      #-1,D3
                LEA.L       NEGRITA,A1
                @DOS        IO_SSTR6,3

                MOVE.L      #7,D2
                MOVEQ       #-1,D3
                LEA.L       BITIMAGE,A1     pone el 'bitimage mode'
                @DOS        IO_SSTR6,3
                MOVE        #256,D5

BUCLE1
                SUBQ        #1,D5
                MOVE.L      D5,D4
                MULU        #128,D4         Calcula la posicion
                MOVE.L      D4,A3            del byte a mandar
                ADDA.L      D6,A3
                ADDA.L      A4,A3
                MOVE.B      (A3),D1
                MOVEQ       #-1,D3
                @DOS        IO_SBYTE,3      envia los bytes...
                @DOS        IO_SBYTE,3
                @DOS        IO_SBYTE,3      por triplicado
                CMP.B       #0,D5
                BNE.S       BUCLE1
                MOVE.L      #13,D2
                LEA.L       MEDIO,A1
                @DOS        IO_SSTR6,3
                MOVE        #256,D5
```

```
BUCLE2
                SUBQ        #1,D5
                MOVE.L      D5,D4
                MULU        #128,D4
                MOVE.L      D4,A3
                ADDA.L      D6,A3
                ADDA.L      A4,A3
                ADDQ        #1,A3
                MOVE.B      (A3),D1
                MOVE.W      #-1,D3
                QDOS        IO_SBYTE,3      y recalca los blancos
                QDOS        IO_SBYTE,3
                QDOS        IO_SBYTE,3
                CMP.B       #0,D5
                BNE.S       BUCLE2
                MOVEQ       #1,D2
                LEA.L       LF,A1
                MOVEQ       #-1,D3
                QDOS        IO_SSTRG,3
                ADD         D2,D6
                CMP.B       #128,D6         acabamos?
                BNE         MANDAR          no!
FINAL
                FIN                         si!
BUSY
                MOVEQ       #36,D2          Si no se puede abrir canal
                MOVEQ       #-1,D3
                LEA.L       OCUPADO,A1      avisar del error
                QDOS        IO_SSTRG,3
                BRA.S       FINAL           y finalizar
*
OCUPADO         DC.B        'Error, canal de la impresora ocupado'
LF              DC.B        10
NEGRITA         DC.B        27,71
PERIFERICO      DC.W        4
                DC.B        'SER1'
BITIMAGE        DC.B        27,65,8,27,76,0,3
MEDIO           DC.B        27,72,27,51,1,10
                DC.B        27,65,8,27,76,0,3
REINICIO        DC.B        27,64
*
                END
```



Javier Boira.

INTRODUCCION AL CODIGO MAQUINA

INSTRUCCIONES DEL 68000

INSTRUCCIONES ARITMETICAS

-ADD

Suma el operando fuente y el destino. El resultado se almacena en el destino.

Ej: ADD (A3),D2

En esta instrucción siempre debe usarse al menos un registro de datos (como operando fuente o destino).

Si el operando destino es un registro de datos, el operando fuente puede indicarse con cualquier modo de direccionamiento:

INSTRUCCION ADD

Modos posibles	op. fuente	op. destino
Reg. datos directo	x	x
Reg. direc. directo	x	
Reg. direc. indirecto	x	
Reg. indirecto con postincr.	x	
Reg. indirecto con predecr.	x	
Reg. indirecto con desplaz.	x	
Reg. indirecto indexado	x	
Absoluto corto	x	
Absoluto largo	x	
Relat. cont. programa con despl.	x	
Relat. cont. programa indexado	x	
Inmediato	x	

En el caso de que funcione un registro de datos como operando fuente no puede indicarse el destino usando direccionamiento relativo al contador de programa, ni direccionamiento inmediato:

INSTRUCCION ADD

Modos posibles	op. fuente	op. destino
Reg. datos directo	x	x
Reg. direc. directo		x
Reg. direc. indirecto		x
Reg. indirecto con postincr.		x
Reg. indirecto con predecr.		x
Reg. indirecto con desplaz.		x
Reg. indirecto indexado		x
Absoluto corto		x
Absoluto largo		x
Relat. cont. programa con despl.		
Relat. cont. programa indexado		
Inmediato		

En el ejemplo expuesto al principio, se suma el contenido de la dirección de memoria apuntada por A3 con D2, almacenando el resultado en D2

-- Alteraciones de los flags

'C' y 'X' Indican si se ha producido un acarreo en el resultado, poniéndose a '1'

'Z' Se pone a '1' si el resultado de la suma es 0, en caso contrario su estado estado es '0'

'N' Presenta '1' si el resultado es negativo y '0' en caso contrario

'V' Se pone a 1 si se ha producido un desbordamiento (al sumar 2 numeros con igual signo el resultado no puede expresarse en complemento a 2 por desbordar el tamaño de operando que tiene asignado, produciéndose un cambio de signo en el resultado)j en caso contrario el flag se pone a '0'

Casos especiales de ADD

-ADDA

Suma un operando fuente que puede indicarse con cualquier modo de direccionamiento a un registro de direcciones.

Ej: ADDA D2,A6

Suma los 16 bits de menos peso de D2 con los respectivos de A6, depositando el resultado en A6, que extiende el signo del resultado (bit de más peso) a los 32 bits.

Debe recordarse que no puede usarse esta instrucción ni ninguna que implique el uso de un registro de instrucciones con datos de 8 bits.

INSTRUCCION ADDA

Modos posibles	op. fuente	op. destino
Reg. datos directo	x	
Reg. direc. directo	x	x
Reg. direc. indirecto	x	
Reg. indirecto con postincr.	x	
Reg. indirecto con predecr.	x	
Reg. indirecto con desplaz.	x	
Reg. indirecto indexado	x	
Absoluto corto	x	
Absoluto largo	x	
Relat. cont. programa con despl.	x	
Relat. cont. programa indexado	x	
Inmediato	x	

La diferencia entre ADDA y ADD con un registro de direcciones estriba en que ADDA no altera ningún flag del registro de estado

-ADDI

Suma un dato inmediato (va junto con la instrucción) a un operando destino que puede indicarse según los modos de direccionamiento:

INSTRUCCION ADDI

Modos posibles	op. fuente	op. destino
Reg. datos directo		x
Reg. direc. directo		
Reg. direc. indirecto		x
Reg. indirecto con postincr.		x
Reg. indirecto con predecr.		x
Reg. indirecto con desplaz.		x
Reg. indirecto indexado		x
Absoluto corto		x
Absoluto largo		x
Relat. cont. programa con despl.		
Relat. cont. programa indexado		
Inmediato	x	

Ej: ADDI #$2B21,(A3)

La instrucción del ejemplo suma el número hexadecimal 2B21 al contenido del la dirección de memoria apuntada por A3, almacenando el resultado en dicha dirección.

-- Alteraciones de los flags
Los flags 'X','N','Z','V','C' se alteran del mismo modo que en la instrucción ADD

-ADDQ

Esta instrucción se denomina "suma rápida", por ejecutarse en menos tiempo que ADD, pero restringe las posibilidades, ya que el operando fuente es un dato inmediato contenido en la misma palabra de la instrucción y su valor está comprendido entre 1 y 8 ya que se expresa con 3 bites.

Los modos de direccionamiento posibles son

INSTRUCCION ADDQ

Modos posibles	op. fuente	op. destino
Reg. datos directo		X
Reg. direc. directo		X
Reg. direc. indirecto		X
Reg. indirecto con postincr.		X
Reg. indirecto con predecr.		X
Reg. indirecto con desplaz.		X
Reg. indirecto indexado		X
Absoluto corto		X
Absoluto largo		X
Relat. cont. programa con despl.		
Relat. cont. programa indexado		
Inmediato	X	

Ej: ADDQ.B #4,$3BC

En el ejemplo se usa direccionamiento absoluto corto, para indicar la dirección de memoria destino, con la que se suma el dato inmediato (4), y donde se almacena el resultado

-- Alteraciones de los flags
Los flags 'X','N','Z','V','C' se alteran del mismo modo que en la instrucción ADD

--ADDX

Suma el operando fuente con el destino, y con el flag 'X', depositando el resultado en el destino.

Ej: ADDX.B D2,D4

El ejemplo lleva a cabo: D2 + D4 + 'X'

--Alteraciones de los flags

'C' y 'X' Se ponen a 1 si se genera acarreo al realizar la operación; en caso contrario se ponen a 0.

'Z' Se pone a "0" si el resultado no es 0; en caso contrario, no cambia.

'N' Se pone a 1 si el resultado es negativo (sólo tiene sentido en números con signo.

'V' Se pone a "1" si se produce un desbordamiento; en caso contrario toma el valor "0"

Esta instrucción suele usarse para comprobar el resultado final en operaciones de múltiple precisión. Antes de llevar a cabo la operación, se suele poner a "1" el flag 'Z' (MOVE to CCR); si al hacer la operación alguna parte del resultado no es cero, 'Z' se pondrá a "0", en caso contrario permanecerá a "1"

Para direccionar los operandos ADDX permite dos posibilidades:

-Suma entre registros de datos: ADDX Dx, Dy

-Suma entre direcciones: ADDX -(Ax),-(Ay) -unicamente es posible direccionamiento por registro de direcciones indirecto con predecremento

-ABCD

Al igual que ADDX, suma el operando fuente, con el destino y con el flag 'X'. El resultado se almacena en el destino. La suma se lleva a cabo en aritmética "BCD" (v. GLave n.1)

Los operandos pueden direccionarse de dos formas:

-Suma entre registros de datos: ABCD Dx, Dy

-Suma entre direcciones: ABCD -(Ax), -(Ay)

ABCD sólo trabaja con operandos de un byte; por lo que la suma entre registros sólo afecta a los 8 bits de menos peso de los mismos, y la suma entre direcciones se lleva a cabo con octetos.

Nótese que la suma entre direcciones sólo puede hacerse usando direccionamiento por registro de direcciones indirecto con predecremento.

--Alteraciones de los flags

'X' y 'C' Se ponen a "1" si se genera un acarreo decimal, si no, pasan a "0"

'Z' Se pone a "0" si el resultado es distinto de cero; en caso contrario no cambia.

'N' y 'V' quedan en estado desconocido

-SUB

Resta en binario el contenido del operando fuente del destino, almacenando el resultado en el destino.

Al igual que ocurre con la instrucción ADD, debe usarse al menos un registro de datos (bien como operando fuente o como destino).

Si es un registro de datos el que provee uno de los operandos y es el destino de la resta, puede emplearse cualquier modo de direccionamiento para especificar el operando fuente :

INSTRUCCION SUB

Modos posibles	op. fuente	op. destino
Reg. datos directo	X	X
Reg. direc. directo	X	
Reg. direc. indirecto	X	
Reg. indirecto con postincr.	X	
Reg. indirecto con predecr.	X	
Reg. indirecto con desplaz.	X	
Reg. indirecto indexado	X	
Absoluto corto	X	
Absoluto largo	X	
Relat. cont. programa con despl.	X	
Relat. cont. programa indexado	X	
Inmediato	X	

Si empleamos un registro de datos para facilitar el operando fuente, los modos de direccionamiento posibles para el destino son:

INSTRUCCION SUB

Modos posibles	op. fuente	op. destino
Reg. datos directo	x	
Reg. direc. directo		
Reg. direc. indirecto		x
Reg. indirecto con portincr.		x
Reg. indirecto con predecr.		x
Reg. indirecto con desplaz.		x
Reg. indirecto indexado		x
Absoluto corto		x
Absoluto largo		x
Relat. cont. programa con despl.		
Relat. cont. programa indexado		
Inmediato		

Ej: SUB (A3)+,D5

En el ejemplo se usa direccionamiento indirecto con predecremento. El resultado de la resta entre el contenido de la direccibn (A3), y el registro D5 se almacena en D5.

--Alteraciones de los flags

'C' Se pone a "1" si se genera una "llevada" en la resta, en caso contrario pasa a "0"

'X' Se comporta igual que 'C'

'Z' Se pone a "1" si el resultado es cero y a "0" en caso contrario.

'N' Se pone a "1" si el resultado es negativo, y a "0" en caso contrario

'V' Pasa a 1 si se produce un desbordamiento (al restar dos números de signos contrarios, no puede contenerse el resultado en el tamaño de complemento a 2).

Casos especiales de SUB

-SUBA

En este caso el operando destino, que facilita el minuendo de la operación, y almacena el resultado es un registro de direcciones.

El operando fuente puede señalarse con cualquier modo de direccionamiento:

INSTRUCCION SUBA

Modos posibles	op. fuente	op. destino
Reg. datos directo	X	X
Reg. direc. directo	X	
Reg. direc. indirecto	X	
Reg. indirecto con postincr.	X	
Reg. indirecto con predecr.	X	
Reg. indirecto con desplaz.	X	
Reg. indirecto indexado	X	
Absoluto corto	X	
Absoluto largo	X	
Relat. cont. programa con despl.	X	
Relat. cont. programa indexado	X	
Inmediato	X	

SUBA no altera ningún flag del registro de estado

-SUBI

Es un caso muy similar a ADDI; resta un dato inmediato, que acompaña a la instrucción del operando destino. El resultado se almacena en el destino. Puede trabajar con bytes, palabras o palabras largas.

Los modos de direccionamiento permitidos son:

INSTRUCCION SUBI

Modos posibles	op. fuente	op. destino
Reg. datos directo		x
Reg. direc. directo		
Reg. direc. indirecto		x
Reg. indirecto con postincr.		x
Reg. indirecto con predecr.		x
Reg. indirecto con desplaz.		x
Reg. indirecto indexado		x
Absoluto corto		x
Absoluto largo		x
Relat. cont. programa con despl.		
Relat. cont. programa indexado		
Inmediato	x	

--Alteraciones de los flags
De la misma manera que en la instrucción SUB

-SUBX

Resta el operando fuente y el flag 'X' del operando destino. El resultado se almacena en el destino. En esencia, opera de igual manera que ADDX, sólo que en este caso restando. También permite únicamente las dos posibilidades de direccionamiento que se señalaban para ADDX.

-- Alteraciones de los flags

'C' y 'X'	Pasan a 1 si se genera una "llevada" en la resta. En caso contrario se ponen a "0"
'Z'	Se pone a "0" si el resultado no es cero; en caso contrario no varía
'N'	Se pone a "1" si el resultado es negativo, y a "0" en caso contrario
'V'	Pasa a "1" si se produce un sobrepasamiento, y a "0" en caso contrario.

-SUBQ

Es una instrucción muy similar a ADDQ, sólo que en este caso en vez de llevar a cabo una suma, se realiza una resta. La resta se hace entre un sustraendo que va junto con la operación y que puede ser un número del 1 al 8 y un operando destino que puede direccionarse con los siguientes modos:

INSTRUCCION SUBQ

Modos posibles	op. fuente	op. destino
Reg. datos directo		x
Reg. direc. directo		x
Reg. direc. indirecto		x
Reg. indirecto con postincr.		x
Reg. indirecto con predecr.		x
Reg. indirecto con desplaz.		x
Reg. indirecto indexado		x
Absoluto corto		x
Absoluto largo		x
Relat. cont. programa con despl.		
Relat. cont. programa indexado		
Inmediato	x	

La ventaja que presenta esta instrucción sobre SUB cuando se trata de restar un número comprendido entre 1 y 8, estriba en la rapidez con que opera SUBQ.

-- Alteraciones de los flags.
Del mismo modo que con la operacida SUB.

Juan Palacio

FIRMWARE

Firmware: QLFLP UPGRADE

Editor: QJUMP Ltd.

Dirección: Cambridge CB4 4QD,England

Precio: 19.95 £ (IVA incluido)

El QLFLP Upgrade se trata de una EPROM de 8k. Su instalación requiere una mínima intervención de hardware, ya que sólo hay que intercambiar la EPROM de origen por la nueva en el interface de Micro Peripherals / Sinclair y para ello ni siquiera hay que emplear el soldador ya que la EPROM se inserta simple y facilmente en la tarjeta.

Como creo que son realmente de interès general los resultados de los tests realizados por QJUMP al interface de Sinclair, en vez de comentar en este número las nuevas utilidades del QLFLP y la falta de algunas otras (por otra parte enmendable por el QL TOOLKIT) voy a dar a conocer dichos resultados y queda asi el comentario propiamente dicho para un próximo número.

Los tests que se realizaron con el Interface de Sinclair mostraron que éste es significativamente más lento que los productos de la competencia.

Las unidades empleadas fueron una CST Qdisc V1.08, de principios de Marzo de 1985, una Micro Peripherals disk Interface V5.3, de Noviembre de 1985, y un QFLP upgrade V1.15. Los tests se realizaron usando un QL con ROM versión JM, ningún otro periférico más que los Interfaces de discos y un par de TEAC FD55F direct drive disk drives.

El QLFLP upgrade no es más que el mismo software usado por todas las unidades de la competencia, configurado para el interface de Micro Peripheral. Tiene todo el usual Wild Card para el manejo de directorios, acceso directo al manejo de archivos y facilidades para el control de jobs.

La seguridad al archivar y la velocidad de acceso es seleccionable por el usuario. Para hacer una comparación razonable entre el upgrade y la unidad original, se realizaron algunos tests para comprotar la robustez del sistema, provocando diversos errores.

1) Un archivo se abre para leer, y mientras permanece abierto se cambia el disco y se abre otro archivo.

2) Un archivo se abre para escribir, y mientras permanece abierto se cambia el disco y se abre otro archivo.

3) Un archivo se abre para leer, y mientras permanece abierto se cambia el disco, después se hace otro intento de leer.

4) Un archivo se abre para escribir, y mientras permanece abierto se cambia el disco después se hace otro intento de escribir.

5) Un archivo se abre para leer, y mientras se està leyendo se cambia el disco.

El QLFLP upgrade fue testeado en los niveles de extrema seguridad de 2 (por defecto) y 0 (la menor seguridad).

En el nivel de seguridad 0 el QLFLP upgrade fallò en todas las pruebas excepto en el test 5.

En el nivel de seguridad 2 el QLFLP se restablece en todos los tests excepto en el 4. En este caso el disco no se vicib, pero los datos del archivo se perdieron, y se produjo un mensaje de advertencia.

La unidad de Sinclair pasò los tests de lectura 1 y 3, aunque no se produjeron mensajes de aviso para indicar al usuario que el archivo del disco anterior permanecia abierto. En los tests de escritura 2 y 4 no dib ninguna indicaciòn de que algo estaba fuera de lugar, el nuevo disco no aparecib viciado, pero los datos de los archivos se desvanecieron. El test 5 fuè tan dramàtico que el QL sufriò un reset.

Si la seguridad de los datos es muy importante, entonces el contenido de los buffers deberia ser nivelado periòdicamente. La unidad de Sinclair no tiene una completa operaciòn de nivelado y un archivo escrito y nivelado, pero no cerrado, se pierde totalmente al remover el disco. El QLFLP upgrade tiene un nivelado completo.

En la prèctica la seguridad de la unidad de Sinclair està situada entre los niveles de seguridad 0 y 2 del QLFLP upgrade.

Mientras se realizaron los tests de seguridad, se encontrò que la unidad de Sinclair no notaba si un disco I estaba protegido de escritura o no !

Es importante que al realizar cualquier actuaciòn en un banco de pruebas sea sencilla y a la vez real, extensa a la vez que compacta, y reproducible.

Los tests deberian distinguir entre las diferentes contribuciones al tiempo tomado en la realizaciòn de las operaciones.

Todos los tiempos se midieron usando el reloj de tiempo real del propio QL. En el caso de operaciones individuales de medida del orden del segundo, se realizaron 60 veces, y se tomò como resultado el promedio de los valores obtenidos obteniendo asi un error en la medida del orden de 0.05 segundos.

El tiempo necesitado para realizar una operaciòn depende de si el disco està corriendo, de la cantidad de memoria libre, del tamaño del directorio y de la posiciòn del archivo en el disco. Por esta razòn, muchos de los tests se repitieron bajo diferentes condiciones.

Cuando fuè posible los tests se realizaron usando programas en BASIC. Hay tres programas en BASIC (test_1_2, test_3_4 y test_5_6_7) y un programa ejecutable para medidas de sucesos muy ràpidos (test_0_bin).

Antes de realizar cada serie de tests, el disco de prueba (protegido de escritura) se pone en el drive 1 y el QL sufre un reset. Despuès se escribe el comando para comenzar el test:

```
WSET: LRUN test_1_2                ( Micro Peripherals )
LRUN test_1_2                      ( QLFLP por defecto )
FLP_opt 0: LRUN test_1_2           ( QLFLP seguridad 0 )
```

TEST GRUPO 0

Se trata al más al menos que del tiempo tomado para formatear un disco de 80 pistas de doble cara.

TEST GRUPO 1 y 2

Es un test sobre el manejo de directorios: búsqueda, creación y borrado de archivos. Los archivos están vacios asi que el tiempo de escritura de los mismos no influye en el resultado. Los tests se hicieron con los drives parados (tests individuales). Se hicieron con y sin archivos abiertos, con discos casi vacios y casi llenos. La totalidad de los tests se repitió con la RAM vacia y llena (éste es el estado normal cuando se usan los programas de PSION y esto posibilita que los tests se realicen con el interface operando con muy poco espacio de buffer).

Estos archivos abiertos y cerrados son particularmente importantes y el tiempo de respuesta al abrir o al fallar en la apertura de un archivo es critico en el uso de sistemas interactivos.

TEST GRUPOS 3 y 4

Estos tests chequean la actuación del tamaño de las operaciones de carga y grabación. Los tests se llevaron a cabo con archivos pequeños (1 Kbyte), medianos (4 Kbytes) y grandes (32 kbytes). El almacenamiento del disco es fragmentado por el borrado variable de pequeños archivos, y la grabación y carga de los archivos grandes es repetida.

Los tests se realizaron con RAM llena y vacia.

TEST GRUPOS 5 y 6

Muchas operaciones requieren que se lea un archivo, mientras se crea o enmienda otro. La simple forma de estas múltiples series de operaciones con archivos es la simple copia de archivos.

Estos archivos de prueba (de diferentes tamaños) son copiados en el mismo disco, y desde un disco a otro. Como hay alguna variación en el posicionamiento relativo de los archivos fuente y destino, los archivos no se copian en el mismo orden en que se crearon.

De nuevo, los tests se realizaron con RAM llena y vacia.

TEST GRUPO 7

Un archivo de 48 Kbytes se copia en y desde un cartucho de Microdrive. Las medidas de tiempos de este test fueron inconsistentes, se tomaron las medidas más rápidas correspondientes a la segunda pasada. Para este test fue necesaria una gran cantidad de memoria.

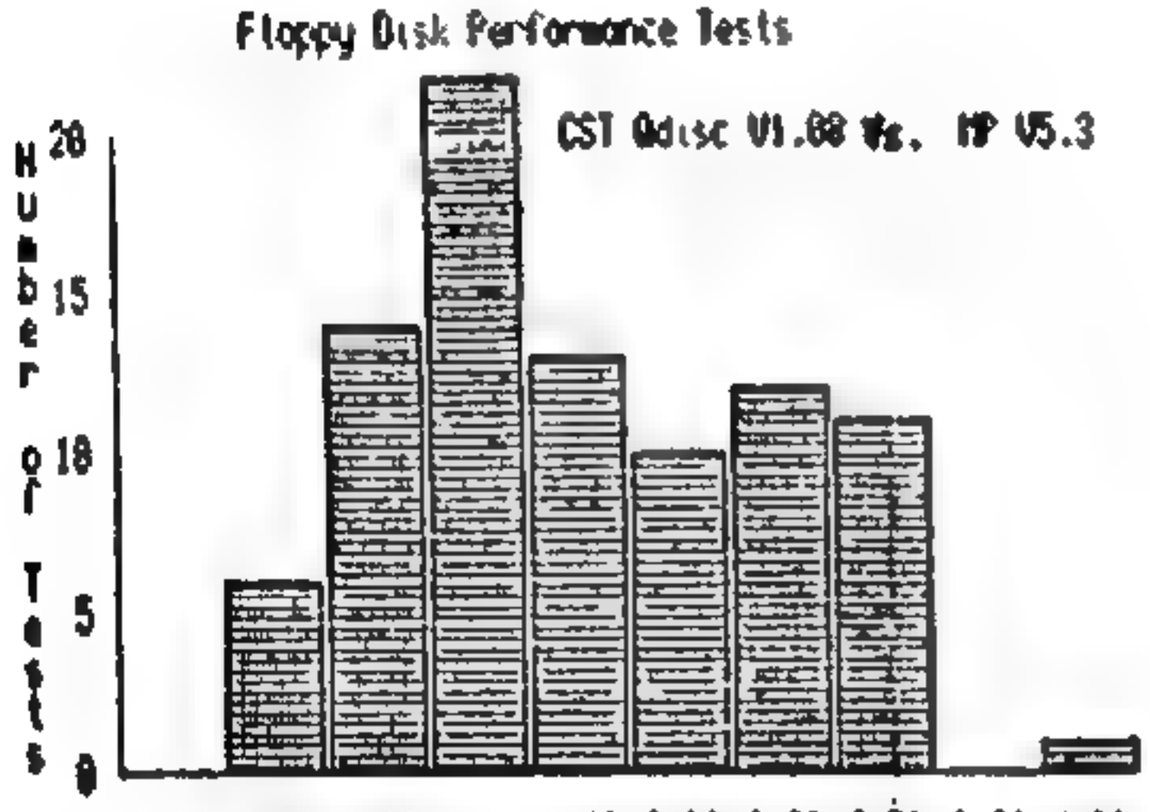
TEST GRUPO 8

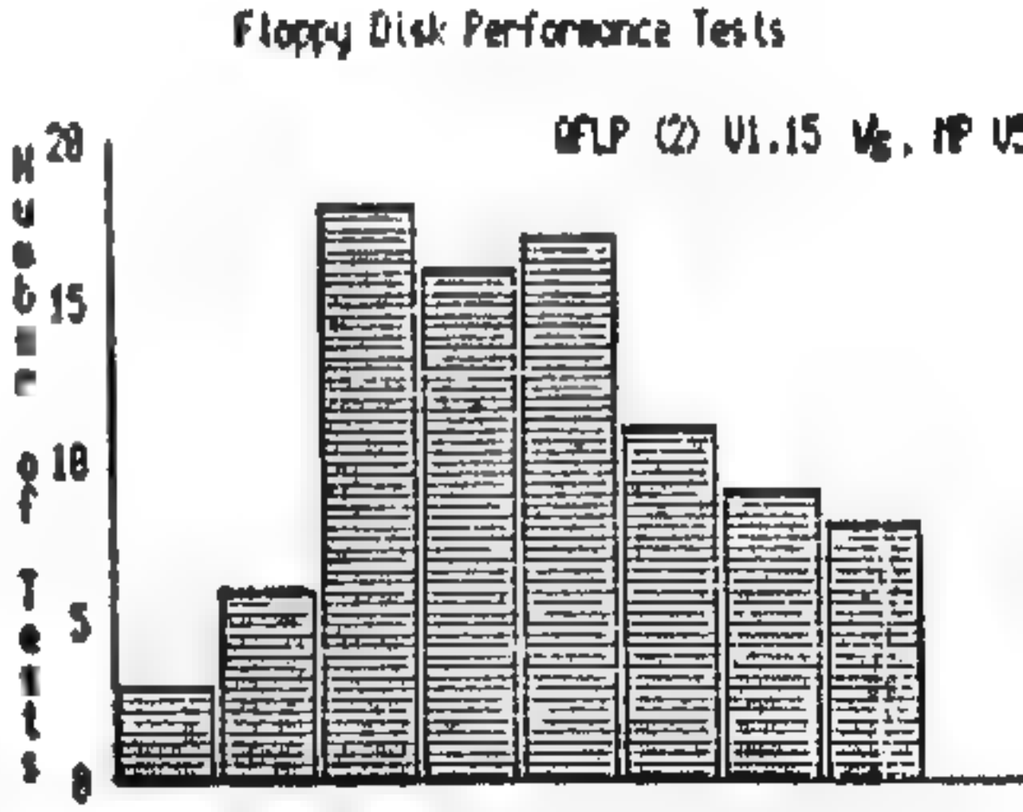
Este grupo de tests está escrito en ensamblador. Primero se calcula la respuesta de la lectura, y después, estando un archivo grande en el buffer, se mide el tiempo para un byte, línea y cadena respectivamente. Finalmente se mide el tiempo de lectura aleatoria de bytes. Si 888 nos parece un largo tiempo para obtener un byte, considerar que el tiempo tomado en llamar a una subrutina y volver con todos los registros grabados es del orden de 148.

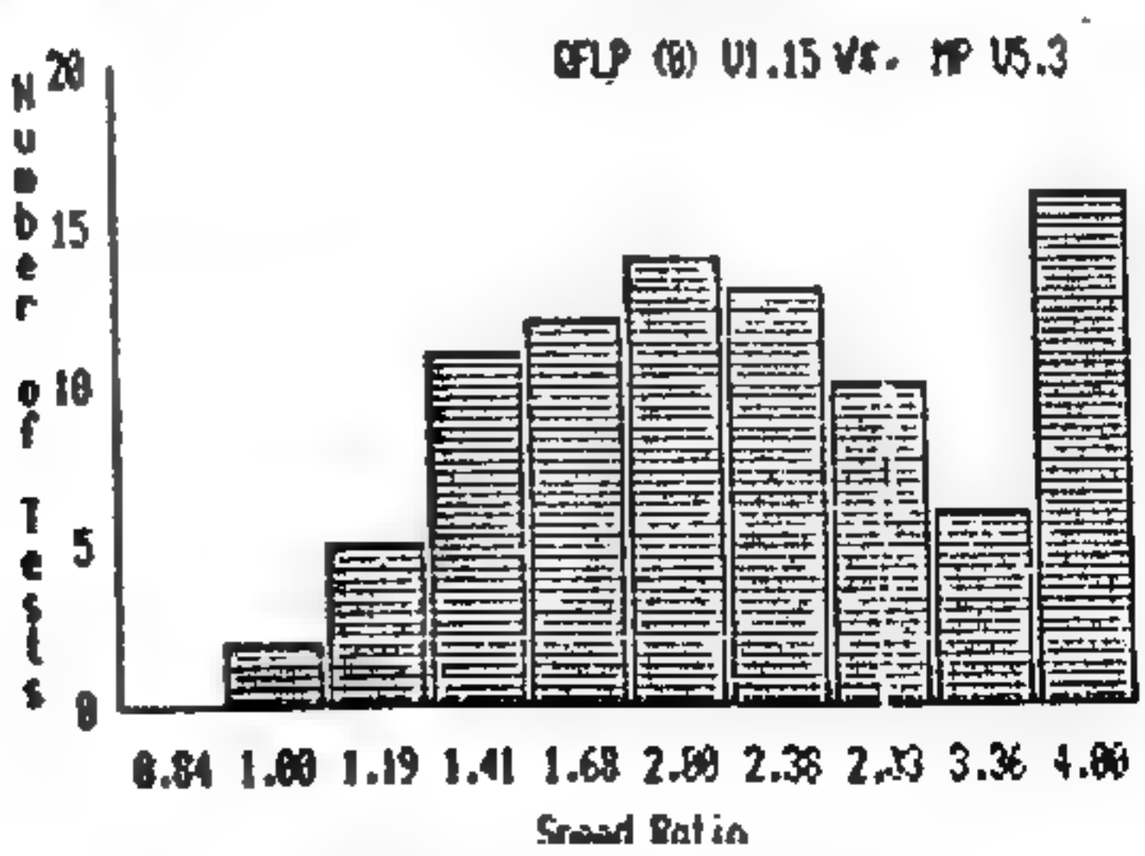
LOS RESULTADOS

Los resultados aparecen tabulados seguidamente, y debido a que los promedios son bastante significativos, únicamente éstos se reproducen en forma de histograma.

Los resultados muestran que el CST 8disc de Marzo de 1985 es bastante mejor que el de SINCLAIR de Noviembre de 1985. Con el 8LFLP upgrade usando los valores de los parámetros por defecto y sin ninguna modificación de hardware, la unidad de SINCLAIR es casi tan rápida como la de CST. Con el nivel de seguridad en 8, la unidad es considerablemente más rápida (aunque no tanto como otros interfaces usando el software de 8JUMP v1.14 o v1.15 - los resultados no se presentan).







Floppy disck Perfonance Tests

		MP V3.3	CST	V1.08	QFLP2 V1.19		QFLPO V1.15	
Test Grupo 0		SECS.	secs.	ratio	secs.	ratio	secs.	ratio
format								
Test 0:	Format 80 track double sided	130.00	109.00	1.19	84.00	1.55	84.00	1.55
Test Grupo 1								
small files: creation and deletion,RAM empty								
Disk emptyo								
Test 1 :	OPEN individual files	1.28	0.55	2.33	0.82	1.56	0.87	1.47
Test 2 :	CLOSE individual files	1.45	1.12	1.29	1.15	1.26	0.02	####
Test 3 :	OPEN individual (file open)	1.75	0.73	2.40	0.93	1.88	0.22	7.95
Test 4 :	CLOSE individual (file open!)	1.43	1.22	1.17	1.22	1.17	0.02	####
Test 5 :	OPEN/CLOSE multiple files	2.18	1.28	1.70	1.28	1.70	0.54	4.04
Test 6 :	OPEN/CLOSE multiple (file open)	2.60	1.48	1.76	1.48	1.76	0.66	3.94
Disk more full								
Test 7 :	OPEN individual files	3.03	1.12	2.71	1.28	2.37	1.30	2.33
Test 8 :	CLOSE individual files	1.90	1.33	1.43	1.98	0.96	0.00	####
Test 9 :	OPEN individual (file open)	3.38	1.30	2.60	1.53	2.21	0.83	4.07
Test 10:	CLOSE individual (file open!)	2.05	1.38	1.49	2.10	0.98	0.00	####
Test 11:	OPEN/CLOSE multiple files	4.18	2.28	1.83	2.32	1.80	1.14	3.67
Test 12:	OPEN/CLOSE multiple (file open)	4.56	2.56	1.78	2.60	1.75	1.70	2.68
At star of disc								
Test 13:	DELETE individual files	1.53	1.47	1.04	1.67	0.92	0.92	1.66
Test 14:	DELETE individual (file open)	1.47	1.50	0.98	1.67	0.88	0.22	6.68
Test 15:	DELETE multiple files	1.46	1.22	1.20	1.22	1.20	0.44	3.32
Test 16:	DELETE multiple files (file open)	1.42	1.20	1.18	1.20	1.18	0.42	3.38
At end of disK								
Test 17:	DELETE individual files	2.75	2.18	1.26	2.40	1.15	1.38	1.99
Test 18:	DELETE individual (file open)	2.70	2.20	1.23	2.43	1.11	0.70	3.86
Test 19:	DELETE multiple files	2.70	2.18	1.24	2.18	1.24	1.10	2.45
Test 20:	DELETE multiple files (file open)	2.64	2.18	1.21	2.18	1.21	1.12	2.36

Test Grupo 2

small files: creation and deletion,RAM full

Disk emptyo

Test 1 :	OPEN individual files	2.08	0.92	2.26	1.17	1.78	1.08	1.97
Test 2 :	CLOSE individual files	1.67	1.12	1.49	1.13	1.48	0.02	####
Test 3 :	OPEN individual (file open)	3.98	1.43	2.78	1.83	2.17	1.55	2.57
Test 4 :	CLOSE individual (file open!)	1.73	1.22	1.42	1.12	1.54	0.00	####
Test 5 :	OPEN/CLOSE multiple files	6.18	2.38	2.60	2.38	2.60	2.08	2.97
Test 6 :	OPEN/CLOSE multiple (file open)	8.08	2.96	2.73	2.96	2.73	2.66	3.04

Disk more full

Test 7 :	OPEN individual files	9.62	3.38	2.85	3.52	2.73	3.60	2.67
Test 8 :	CLOSE individual files	2.58	1.33	1.94	2.02	1.28	0.00	####
Test 9 :	OPEN individual (file open)	11.50	4.05	2.84	4.30	2.67	4.12	2.79
Test 10:	CLOSE individual (file open!)	2.92	1.48	1.97	2.17	1.35	0.02	####
Test 11:	OPEN/CLOSE multiple files	14.56	5.00	2.91	5.02	2.90	4.74	3.07
Test 12:	OPEN/CLOSE multiple (file open)	16.42	5.60	2.93	5.62	2.92	5.32	3.09

At star of disc

Test 13:	DELETE individual files	2.83	1.85	1.53	2.00	1.42	1.35	2.10
Test 14:	DELETE individual (file open)	1.97	1.85	1.06	2.00	0.99	1.15	1.71
Test 15:	DELETE multiple files	3.74	2.54	1.47	2.54	1.47	2.02	1.85
Test 16:	DELETE multiple files (file open)	3.12	2.32	1.24	2.52	1.24	2.00	1.56

At end of diskK

Test 17:	DELETE individual files	6.83	4.47	1.53	4.65	1.47	3.92	1.74
Test 18:	DELETE individual (file open)	6.00	4.50	1.33	4.67	1.28	3.73	1.61
Test 19:	DELETE multiple files	7.90	5.16	1.53	5.18	1.53	4.69	1.68
Test 20:	DELETE multiple files (file open)	7.46	5.16	1.45	5.16	1.45	4.68	1.59

Test Grupo 3

Load and save files, RAM empty

Test 1 :	SBYTES 1 kbyte files individually	3.27	2.17	1.51	2.52	1.30	1.00	3.27
Test 2 :	SBYTES 4 kbyte files individually	3.83	2.73	1.40	3.00	1.28	1.07	3.50
Test 3 :	SBYTES 32 kbyte files individually	9.10	7.60	1.20	7.90	1.15	6.50	1.40

Test 4 :	LBYTES 1 kbyte files individually	2.03	1.32	1.54	1.62	1.25	1.57	1.29
Test 5 :	LBYTES 4 kbyte files individually	2.58	1.58	1.63	1.82	1.42	1.80	1.43
Test 6 :	LBYTES 32 kbyte files individually	4.30	3.20	1.34	3.60	1.19	3.60	1.19
Test 7 :	SBYTES fragmented 32 kbyte files	9.20	8.00	1.15	8.30	1.11	6.90	1.33
Test 8 :	LBYTES fragmented 32 kbyte files	5.40	4.20	1.29	4.70	1.15	4.70	1.15

Test Grupo 4

oad and save files, RAM full

Test 1 :	SBYTES 1 kbyte files individually	5.85	2.58	2.27	2.85	2.05	1.53	3.82
Test 2 :	SBYTES 4 kbyte files individually	7.02	3.57	1.97	3.78	1.86	3.33	2.11
Test 3 :	SBYTES 32 kbyte files individually	14.80	10.00	1.48	10.40	1.42	9.50	1.56
Test 4 :	LBYTES 1 kbyte files individually	3.50	1.83	1.91	1.98	1.77	2.10	1.67
Test 5 :	LBYTES 4 kbyte files individually	4.27	2.03	2.10	2.45	1.74	2.27	1.88
Test 6 :	LBYTES 32 kbyte files individually	5.90	3.70	1.59	4.10	1.44	4.20	1.40
Test 7 :	SBYTES fragmented 32 kbyte files	16.80	12.00	1.40	12.40	1.35	11.10	1.51
Test 8 :	LBYTES fragmented 32 kbyte files	5.50	4.30	1.28	4.60	1.20	4.60	1.20

Test Grupo 5

py files RAM empty

Test 1 :	COPY small files (0-2 kbytes)	2.48	1.66	1.70	1.44	1.72	0.88	2.82
Test 2 :	COPY drive to drive	2.40	1.62	1.48	1.52	1.58	0.90	2.67
Test 3 :	COPY again (disk more full)	3.20	1.78	1.81	1.80	1.79	1.20	2.68
Test 4 :	COPY drive to drive again	3.00	1.82	1.65	1.78	1.69	1.04	2.88
Test 5 :	COPY medium files (11-20 kbytes)	11.50	6.80	1.69	6.90	1.67	7.20	1.60
Test 6 :	COPY drive to drive	12.60	7.10	1.77	6.80	1.85	6.50	1.94
Test 7 :	COPY again (disk more full)	14.00	9.20	1.52	9.30	1.51	9.10	1.54
Test 8 :	COPY drive to drive again	12.90	7.10	1.82	7.10	1.82	7.10	1.82
Test 9 :	COPY large files (40-70 kbytes)	48.25	23.25	2.08	24.50	1.97	23.75	2.03
Test 10:	COPY drive to drive	46.75	20.00	2.34	20.00	2.34	19.75	2.37
Test 11:	COPY again (disk more full)	49.75	32.50	1.53	34.00	2.46	33.75	1.47
Test 12:	COPY drive to drive again	48.50	21.25	2.28	21.50	2.26	21.25	2.28

Test Grupo 6

Copy files RAM full

Test 1 :	COPY small files (0-2 kbytes)	7.38	2.72	2.71	2.70	2.73	3.02	2.4
Test 2 :	COPY drive to drive	5.72	2.52	2.25	2.60	2.20	2.48	2.3
Test 3 :	COPY again (disk more full)	9.57	3.54	2.70	2.54	2.70	3.84	2.5
Test 4 :	COPY drive to drive again	7.38	3.34	2.21	2.34	2.21	3.20	2.3
Test 5 :	COPY medium files (11-20 kbytes)	33.50	15.50	2.16	15.80	2.12	15.60	2.1
Test 6 :	COPY drive to drive	25.40	11.00	2.15	11.30	2.25	11.20	2.2
Test 7 :	COPY again (disk more full)	41.70	18.90	2.21	20.20	2.06	20.20	2.0
Test 8 :	COPY drive to drive again	21.80	13.20	2.03	13.40	2.00	13.20	2.0
Test 9 :	COPY large files (40-70 kbytes)	112.75	51.50	2.19	53.00	2.13	51.75	2.1
Test 10:	COPY drive to drive	88.75	35.25	2.29	33.25	2.43	33.00	2.4
Test 11:	COPY again (disk more full)	148.50	65.25	2.28	66.75	2.22	69.75	2.1
Test 12:	COPY drive to drive again	81.25	40.50	2.01	44.00	1.93	40.50	2.0

Test Grupo 7

Copying to and Microdrive

Test 1 :	COPY to Microdrive	1198.00	42.00	####	30.00	####	35.00	###
Test 2 :	COPY from Microdrive	46.00	43.00	1.07	51.00	0.90	42.00	1.1

Test Grupo 8

Read response and buffering

Test 1 :	Read response, file at end	1.70	1.06	1.60	1.09	1.56	0.72	2.3
Test 2 :	Read response, file at star	0.96	0.69	1.39	0.93	1.03	0.67	1.4
Test 3 :	Read 1000 bytes	0.80	0.78	1.03	0.80	1.00	0.80	1.0
Test 4 :	Read 500 lines (37 characters)	1.26	0.94	1.34	0.94	1.34	0.94	1.3
Test 5 :	Read 500 strings (50 bytes)	1.58	1.14	1.39	1.14	1.39	1.14	1.3
Test 6 :	Position and fetch byte 1000 times	3.20	3.20	0.99	3.24	0.99	3.26	0.9

EL RATIO ESTA CALCULADO
TOMANDO EL VALOR DE LA
UNIDAD MP V5.3 DIVIDIDO
POR EL OBTENIDO POR CADA
UNIDAD

CORREO DE LOS SOCIOS

PROGRAMA: MATCHPOINT.

EDITORIAL: PSION (con licencia de D&L Research).

Nos encontramos ante una creación de Psion basada en el mismo juego para Spectrum de la casa D&L Research. El programa lleva la protección que va siendo habitual en esta casa por lo que habrá que colocar el programa original en mdv2_ y una copia en mdv1_.

Una vez efectuada la carga después de una pantalla de presentación nos aparece un menú de selección de nivel de juego y número de jugadores. En cuanto al nivel de juego existen tres diferentes y que varían en dificultad, estos son cuartos de finales, semifinales y finales. En lo que se refiere al número de jugadores tenemos dos opciones diferentes y que son jugar contra el ordenador o jugar contra un adversario humano. Destacar que dispone así mismo de un modo de demostración en cualquiera de los niveles. Una vez introducido el nivel de juego y el número de jugadores el ordenador nos solicita el nombre de los jugadores pulsando a continuación la barra espaciadora pasamos a la pantalla principal del programa.

Esta pantalla, realizada en modo 4 (alta resolución), nos muestra con todo lujo de detalles una pista de tenis donde no faltan los jueces, recoge pelotas, público y por supuesto los jugadores. Así mismo en la parte superior de la pantalla está situado el marcador que nos indica el tanteo del partido en sets, juegos y puntos.

Los movimientos de nuestro jugador se realizan con las flechas que hay en la parte baja del teclado y para golpear la pelota la barra espaciadora. Si utilizamos la opción de dos jugadores el segundo jugador utilizará las teclas de función. Aquí surge el principal problema de este programa y que es el difícil manejo del jugador y aún más del segundo jugador. Para solucionar este problema disponemos de la posibilidad de jugar con joysticks lo que resuelve en gran medida esta dificultad.

Una vez aquí empezamos a disputar el partido, podemos observar la gran calidad del movimiento de los jugadores y la rapidez con la que lo hacen. Destacar que los recoge pelotas en una simpática carrera recogen la pelota cuando ésta pega en la red; así como los jueces situados en la parte trasera del campo levantan la mano cuando la pelota sale fuera del mismo. En la parte baja de la pantalla nos aparecen una serie de mensajes (second serve, game player 1, set ball, out ...) que nos informan de la marcha del partido.

En cuanto a las reglas del juego son la mismas que en el tenis real, así el ganador será el que gane tres set de un total de cinco posibles. Aunque si se juega con el ordenador es difícil ganar algún set ya que el ordenador se las sabe casi todas.

En resumen decir que es un buen juego del que destaca la calidad de los gráficos y la velocidad y del que como defecto podríamos achacarle la dificultad de manejo; también le faltan una serie de detalles incluidos en la versión para Spectrum como es por ejemplo el poder elegir el número de sets del partido.

Miguel Echavarría Martínez.

NOTIFICACIONES

LIBRERIA DE PROGRAMAS

Este mes se han incorporado a la libreria los siguientes programas:

6.- EL ATOM.

7.- 3-BARRAS-3D.

El Atom está compuesto por los siguientes ficheros:

- clone	410 bytes.
- Boot	233 bytes.
- Atom	31455 bytes.
- Atomaturas	241 bytes.
- Atomas	2142 bytes.
- Ayuda	8048 bytes.

Se trata de un programa que según nos describe su autor tiene tres funciones:

Cálculo de tantos por ciento de masas atómicas en una molécula.

Una hoja electrónica aplicada a los cálculos estequiométricos que se realizan en una reaccion.

Cálculo de las relaciones isotópicas de un ión molecular, este cálculo se aplica para interpretar espectros de masas de compuestos que contienen átomos con varios isótopos.

El programa se autocarga en mivi_ y posee un fichero de ayuda accesible pulsando F1.

El programa 3-BARRAS-3D posee una longitud de 25977 bytes.

Se trata de un programa que permite la representación de gráficos de barras en tres dimensiones. Su menú de opciones es el siguiente:

- Cargar datos de mdv_
- Salvar datos en mdv_
- Imprimir el gráfico obtenido en pantalla (Copy)
- Cambiar las barras en curso : Su color, valor, anchura y color de fondo.
- Cambiar el tipo de barra: Cuadradas, rectangulares, semicuadradas, personal.
- Introducción de textos en la pantalla.

Respecto al método de acceso a de envio de programas de la libreria es el mismo que el utilizado en el mes anterior por lo que os remitimos al número del mes anterior.

FLASH . . . FLASH . . .

QL User una de las publicaciones de mayor importancia sobre QL en U.K. ha pasado a incorporarse a la publicación QL World que desde ahora agrupará a las dos publicaciones.

APARTADO DE CORREOS ■ ■ ■ ■ ■ ■

A partir de este mes hemos decidido centralizar las comunicaciones que nos enviéis. Asi culaquier tipo de comunicación, colaboración, etc.. que deseeis enviarnos hacedlo a la siguiente dirección :

APARTADO DE CORREOS 403
50080 ZARAGOZA

Esperamos que con este cambio os animeis a escribir más y nos permita llevar un servicio más eficente de nuestro/vuestro club.

SUMARIO

1 .- PORTADA
2 .- INFORMACION SOBRE EL CLUB
3 .- EDITORIAL
4 .- COMENTARIO DE PROGRAMAS
ALSICONT
8 .- NOVEDADES
11.- PREGUNTAS
14.- TRUCOS Y RUTINAS
DUMP
23.- INTRODUCCION AL C. MAQUINA
33.- FIRMWARE
UPGRADE
41.- CORREO DE LOS SOCIOS
42.- NOTIFICACIONES
44.- SUMARIO